Signal Processing 92 (2012) 2002–2017

Contents lists available at SciVerse ScienceDirect

Signal Processing

journal homepage: www.elsevier.com/locate/sigpro

Poisson image denoising using fast discrete curvelet transform and wave atom

Sandeep Palakkal *, K.M.M. Prabhu

Department of Electrical Engineering, IIT Madras, Chennai 600 036, India

ARTICLE INFO

Article history:
Received 24 August 2011
Received in revised form
26 November 2011
Accepted 7 January 2012
Available online 14 January 2012

Keywords:
Poisson image denoising
Multiscale variance stabilizing transform (MS-VST)
Fast discrete curvelet transform (FDCT)
Wave atom (WA)

ABSTRACT

In this paper, we propose a strategy to combine fast discrete curvelet transform (FDCT) and wave atom (WA) with multiscale variance stabilizing transform (MS-VST); our objective is to develop algorithms for Poisson noise removal from images. Applying variance stabilizing transform (VST) on a Poisson noisy image results in a nearly Gaussian distributed image. The noise removal can be subsequently done assuming a Gaussian noise model. MS-VST has been recently proposed in the literature (i) to improve the denoising performance of Anscombe's VST at low intensity regions of the image and (ii) to facilitate the use of multiscale-multidirectional transforms like the curvelet transform for Poisson image denoising. Since the MS-VST has been implemented in the space-domain, it is not clear how it can be extended to FDCT and WA, which are incidentally implemented in the frequency-domain. We propose a simple strategy to achieve this without increasing the computational complexity. We also extend our approach to handle the recently developed mirror-extended versions of FDCT and WA. We have carried out simulations to validate the performance of the proposed approach. The results demonstrate that the MS-VST combined with FDCT and WA are promising candidates for Poisson denoising.

1. Introduction

In several imaging modalities where image capturing involves detection of particles such as photons, the captured image may be modelled as a realization of a two-dimensional (2-D) Poisson random process [1]. The corresponding mathematical model is

$$x[\mathbf{m}] \sim \text{Poisson}(\lambda[\mathbf{m}]), \tag{1}$$

where $\mathbf{x} = \{x[\mathbf{m}]\}_{\mathbf{m}}$ is the observed image, $\boldsymbol{\lambda} = \{\lambda[\mathbf{m}]\}_{\mathbf{m}}$ is the true image, and $\mathbf{m}$ is the index vector $\mathbf{m} = [m_1\ m_2]$, with $m_i = 0, 1, \ldots, N-1$, for $i = 1,2$. The observed pixel values $\{x[\mathbf{m}]\}_{\mathbf{m}}$ are conditionally independent. That is, given the true image $\boldsymbol{\lambda}$, the pixel values of $\mathbf{x}$ follow independent Poisson distribution. Prior to further processing, it is necessary to denoise the observed image $\mathbf{x}$ by estimating its true intensity profile $\boldsymbol{\lambda}$. In doing this, a major difficulty arises in handling the hetroskedastic nature of noise: from the noise model described in (1), it can be observed that the parameter of the Poisson noise model $\lambda[\mathbf{m}]$ (i.e., the noise variance) depends on the spatial location $\mathbf{m}$. Also, it is straightforward to show that the signal-to-noise ratio (SNR) of the Poisson noisy images is $\lambda[\mathbf{m}]$ at location $\mathbf{m}$.

A popular strategy for Poisson denoising is to transform the noise model from Poisson to Gaussian by using a variance stabilizing transform (VST). Precisely, the output of the VST tends to be Gaussian distributed with a uniform variance (i.e., homoskedastic), as the image intensities approach infinity. Thus, denoising can be performed in the Gaussian domain for which several methods exist in the literature, and the final estimate of the image may be obtained by inverting the VST. The most popular examples for VST are Anscombe's VST [2], and the Haar–Fisz transform [3], which combines the Fisz transform [4] with the Haar wavelet transform. In this paper, we focus on Anscombe's VST [2]. A major disadvantage of

* Corresponding author. Tel.: +91 44 22575471.
E-mail addresses: ee07d013@ee.iitm.ac.in, sandeep.dion@gmail.com (S. Palakkal), prabhu@ee.iitm.ac.in (K.M.M. Prabhu).

0165-1684/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
doi:10.1016/j.sigpro.2012.01.008

Anscombe's VST is that its denoising performance is very poor at low image intensities (i.e., when the SNR is very low) [1,5]. In order to overcome this, a multiscale VST (MS-VST) has been proposed recently [6]. The idea is to concatenate a slightly modified VST after the lowpass filters associated with multiscale transforms, e.g., wavelet transform. The lowpass filter reduces part of the noise, thereby improving the SNR to some extent before passing the image through VST. Furthermore, exploiting the sparsity offered by the multiscale transforms in representing the intrinsic edges of most of the images, very good denoising procedures may be developed, e.g., hard-thresholding [7,8]. As a combined effect, the MS-VST shows very good denoising capability, even at low image intensities [6].

The MS-VST defined in [6] depends on the impulse responses of the associated filters, and is implemented exclusively in the space-domain. Since many of the multiscale-multidirectional transforms are implemented using filter banks, the MS-VST may be easily incorporated into any of them. Particularly in [6], MS-VST was implemented using the undecimated wavelet transform (UWT), ridgelet transform and the first generation curvelet transform (CVT) [8]; among these, CVT shows the best denoising performance [6]. However, there exist multiscale transforms which are implemented in the Fourier domain by applying frequency-domain windows corresponding to each scale and, possibly, directions. Examples of such transforms include the fast discrete curvelet transform (FDCT) [9] and wave atom (WA) [10]. Though such transforms possess an equivalent filter bank structure, it is not clear how to extend the definition of MS-VST to the Fourier domain. The difficulty here is twofold. Firstly, MS-VST [6] is a nonlinear transform involving the square-root function, and therefore, an equivalent operation in the Fourier domain is not straightforward to implement. Secondly, the probability distribution of the Fourier coefficients of a Poisson distributed image has no closed form expression, and hence, the derivation of MS-VST in the Fourier domain is difficult. However, since the use of FDCT and WA have been shown to result in very good denoising techniques [11–13], it is advantageous to develop MS-VST for these transforms as well.

In this paper, we propose a simple and direct method to combine MS-VST with FDCT and WA. We propose to implement MS-VST in the space-domain using a generalized Laplacian pyramid (LP) [14], and combine this structure with the existing FDCT and WA structures. The design of the filters in LP depends on the transform with which we intend to combine MS-VST. We also explain how the proposed strategy can be used with the mirror-extended FDCT (ME-FDCT) and mirror-extended WA (ME-WA). A part of this work along with some preliminary results were presented in [15].

1.1. Related work

Recently, it was observed in [16] that the poor denoising performance of Anscombe's VST at low image intensities is due to the bias introduced by the function used to invert the effect of the VST after Gaussian denoising. In order to overcome this problem, an exact unbiased inverse function for Anscombe's VST (i.e., exact unbiased inverse VST or EU-IVST) was proposed in [16]. Furthermore, it was shown in [16] that the use of EU-IVST along with the recently proposed BM3D algorithm [17] for Gaussian denoising would yield best results. Apart from VST, there are methods which attempt to solve the Poisson denoising problem directly, without transforming the noise model to Gaussian. The most prominent examples include platelet [18] and PURE-LET [19], and the Bayesian methods proposed in [20–22]. More details on these and other methods used for the Poisson denoising problem can also be found in the review papers such as [1,23,24].

1.2. Organization of the paper

The paper is organized as follows. We start by briefly discussing the MS-VST combined with a generalized LP filter bank in Section 2. In Section 3, we propose a structure to combine MS-VST with FDCT. We discuss how a similar structure can be designed for WA in Section 4. Then we extend the proposed scheme to the ME-FDCT and ME-WA in Section 5. To illustrate the denoising performance of the proposed MS-VST structures, we use the same denoising procedure proposed in [6], which we briefly outline in Section 6. We have carried out extensive simulations to validate the denoising performance of the proposed methods. The results are discussed in detail in Section 7. Finally, we conclude the paper in Section 8.

2. Multiscale variance stabilizing transform (MS-VST)

In this section, we briefly explain the MS-VST scheme proposed in [6]. For the sake of the subsequent discussions, we use a generalized oversampled LP filter bank shown in Fig. 1. In Fig. 1, $\mathbf{h}_j$ and $\mathbf{g}_j$, respectively, are the 2-D lowpass and highpass filters in the j-th level of the filter bank, and the block $T_j(\cdot)$ performs the MS-VST operation defined in [6], i.e.,

$$T_j(\mathbf{a}_j) = b_j \operatorname{sgn}(\mathbf{a}_j + c_j)\sqrt{|\mathbf{a}_j + c_j|}, \tag{2}$$

where the symbol sgn(·) represents the signum function. Furthermore, the constants b_j and c_j are defined as

$$b_j = 2\sqrt{|\tau_j^{(1)}|/\tau_j^{(2)}}, \quad c_j = \frac{7\tau_j^{(2)}}{8\tau_j^{(1)}} - \frac{\tau_j^{(3)}}{2\tau_j^{(2)}}, \tag{3}$$

where $\tau_j^{(k)} = \sum_{\mathbf{m}} (h_j[\mathbf{m}])^k$. Then, the outputs of each lowpass filter and highpass filter in Fig. 1 are given, respectively, by

$$\mathbf{a}_j = \mathbf{h}_j * \mathbf{x} \tag{4}$$

and

$$\mathbf{d}_j = \mathbf{g}_j * T_j(\mathbf{a}_j), \tag{5}$$

where $*$ stands for the linear convolution operation.

We conclude this section by stating Theorem 2 from [6], slightly modified for the LP structure given in Fig. 1. Assume that the input image $\mathbf{x}$ is Poisson distributed with parameters $\boldsymbol{\lambda}$ as in (1). Theorem 1 given below states that the output $\mathbf{d}_j$ of the highpass filter $\mathbf{g}_j$ in Fig. 1